Hierarchical differential evolution for minimal cut sets identification: Application to nuclear safety systems

Francesco Di Maio, Samuele Baronchelli, Enrico Zio

HIERARCHICAL DIFFERENTIAL EVOLUTION FOR MINIMAL CUT SETS IDENTIFICATION: APPLICATION TO NUCLEAR SAFETY SYSTEMS

Francesco Di Maio[1], Samuele Baronchelli[1], Enrico Zio[1,2]

[1] *Energy Department, Politecnico di Milano*
Via Ponzio 34/3, 20133 Milano, Italy
francesco.dimaio@polimi.it

[2] *Chair on System Science and Energetic Challenge*
European Foundation for New Energy – Electricite de France
Ecole Centrale, Paris, and Supelec, Paris, France

ABSTRACT

In this paper, we present a Hierarchical Differential Evolution (HDE) algorithm for minimal cut set (mcs) identification of coherent and non-coherent Fault Trees (FTs). In realistic application of large-size systems, problems may be encountered in handling a large number of gates and events. In this work, to avoid any approximation, mcs identification is originally transformed into a hierarchical optimization problem, stated as the search for the minimum combination of cut sets that can guarantee the best coverage of all the minterms that make the system fail: during the first step of the iterative search, a multiple-population, parallel search policy is used to expedite the convergence of the second step of the exploration algorithm. The proposed hierarchical method is applied to the Reactor Protection System (RPS) of a Pressurized Water Reactor (PWR) and to the the Airlock System (AS) of a CANadian Deuterium Uranium (CANDU) reactor. Results are evaluated with respect to the accuracy and computational demand of the solution found.

Keywords: Dynamic Reliability; Minimal Cut Sets (mcs); Fault Trees (FTs); Hierarchical Differential Evolution (HDE).

1. INTRODUCTION

Fault Tree (FT) is a tool widely used in Probabilistic Safety Assessment (PSA) of Nuclear Power Plants (NPPs) [NUREG, 1983; NASA, 2002; Zio, 2007]. Traditionally, FTs are used for quantifying various probabilistic measures (including probabilities and/or frequencies of sequences, safety margins, importance factors and sensitivity indices) [Høyland et al., 1994; Kumamoto et al., 1996; Epstein et al., 2005; Gao et al., 2007; Borgonovo, 2010]. The size of the system may challenge the FT analysis, in practical situations. For example, in the first case study considered in this work for the Reactor Protection System (RPS) of a Pressurized Water Reactor (PWR) with 12 components [Wash-1400, 1976], the minimal cut sets (mcs) identification problem gives rise to a FT structure function Φ composed by 2^{12}=4096 minterms (products of the literals α representing each component state, α=1 failed, α=0 safe), 485361 cut sets (combinations of components failures leading the system into failure) and a cut set chart (table with all minterms as columns and cut sets as rows) of 1966682772 elements.

To overcome the problem, research efforts have developed in two directions: one looking for approximations of the probabilistic measures of interest obtained by considering only some selected mcs; another one developing computational methods to more efficiently assess the probabilistic measures from the exact mcs. One example of approximation consists in considering only small order mcs (i.e., mcs formed by a small number of elements) [Rauzy, 2001], which in principle capture the main part of the top-event probability. Another truncation process selects only the mcs with probability of occurrence larger than a given threshold. However, mcs truncation can have direct consequences on the safety level of the NPP, because it is not known how many are the mcs neglected (because of small order or probability) in the estimation of the risk/safety indicators of interest. For this reason, it has been pointed out that mcs exact identification (rather than truncation) is one of the technical issues to be tackled in the development of PSA for risk-informed decision making, e.g. for maintenance, service inspections and safety margins quantification in new NPPs design [Fleming, 2003; Duflot et., al, 2009; Zio et al., 2010].

A first attempt in developing computational methods for limiting the mcs combinatorial explosion of FTs without approximation has been to encode the Boolean formulae derived by the FTs into binary decision diagrams (BDDs) [Akers, 1978]. One of the major advantages of a BDD is that it provides exact values for probabilistic measures and it does not need any kind of truncation or approximation. However, BDD is highly memory consuming and very large models are beyond capability [Rauzy et al., 1997]. Another attempt for identifying mcs is the Dynamic Flowgraph Methodology (DFM), which is a directed graph-based approach to model and analyze the behavior of dynamic systems [Garrett et al., 1995]. The main drawback is scalability, in that realistic

modeling causes a combinatorial explosion as the number of states in the system increases [Bjorkman, 2013]. In order to tackle this challenge, a DFM has been solved by a BDD (based on meta-products or on zero-suppressed BDD) [Bjorkman, 2013]. Also Petri nets suffer from the combinatorial explosion of the number of states, when applied to complex systems [Labeau et al., 2000].

We propose a novel approach to tackle the issue of exact mcs identification of coherent and non-coherent FTs based on a Hierarchical Differential Evolution (HDE) algorithm. The Differential Evolution (DE) algorithm has been demonstrated to be an efficient, effective, fast and robust method for the identification of prime implicants (PIs) in simple non-coherent structure functions: a comparison with respect to a traditional analytical approach known as Quine-McCluskey algorithm and a Genetic Algorithm (GA) has been presented in [Di Maio et al., 2013]. In the present paper, DE is applied within a hierarchical scheme to deal with its computational limitations and avoid any approximation in the identification of mcs of complex coherent structure functions. With the proposed scheme, we look for the minimum combination of cut sets that can guarantee the best coverage of all the minterms that make the system fail: during the first step of the iteration process, a multiple-population, parallel search policy is implemented to expedite the convergence of the second step of the exploration algorithm.

The paper is organized as follows. Section 2 is devoted to recalling some basic terminology (FT, Boolean Formulae, coherent and non-coherent structure functions, minterms, etc.). In Section 3, the HDE technique for mcs identification is presented. In Section 4.1, it is applied to the FT of a Reactor Protection System (RPS) of a PWR and its results are compared with those obtained with a DE algorithm, whereas in Section 4.2 the results of the application of HDE to the Airlock System (AS) of a CANDU reactor are shown. Conclusions and remarks are given in Section 5.

2. TERMINOLOGY

In this Section, we introduce the terminology used throughout the article with reference to FT analysis. The causal relations that lead to the FT top event can be described by a set of Boolean formulae built over a set of variables (literals) α_1, α_2, α_3,... α_n, and connectives (and, or, not, *k*-out-of-*n*), whose semantics are defined by means of truth tables. By manipulation of the truth tables, the top event can be expressed in terms of the *n* primary events (e.g., components failures in our case of interest). The simplest way to express the structure function Φ, which relates the top event to the primary events, is in terms of minimal cut sets (mcs) $\Pi^* \in \Omega$, where Ω is the whole set of cut sets. A mcs is an irreducible combination of primary events (cut set Π), which if all verified cause the top event to occur. Then, a mcs Π^* is one of the 2^n products of literals (minterms), whose

occurrence ensures the failure of the system $\Phi(\Pi^*)=1$, while no proper subset of Π^* is a cut set [Epstein, 2005]. A structure function Φ is coherent if it can be expressed without any Π^* of complemented literals $\bar{\alpha}$, non-coherent otherwise.

3. A NOVEL TECHNIQUE FOR MCS IDENTIFICATION

We treat the problem of mcs identification as a set covering problem (SCP) [Beasley et al., 1996]. In the context of mcs identification, the SCP is the problem of covering each one of the minterms by a group of cut sets of minimal cost. We define the cost of a cut set Π as the number of literals α associated with system components included in the cut set (literal cost). Within the evolutionary optimization scheme here proposed, each solution of the SCP, $\hat{\bar{x}}_{opt}$, is represented by a specific combination of independent variables, or, mathematically speaking, by a *R*-dimensional vector $\bar{x}=(x_1,x_2,...,x_R)$ (hereafter called chromosome, within the jargon of the DE-optimization method adopted) where a value of 1 in the *i*-th vector position x_i implies that Π_i is chosen to be in the cover and vice versa a value of 0 [Sen, 1993]. The total cost of each possible solution $\hat{\bar{x}}_{opt}$ is defined as the combination of two parts: the literal cost of the cut sets selected to be in the cover and the cost associated with the number of faulty minterms left uncovered by the solution.

3.1 Differential Evolution

DE belongs to the class of Evolutionary Algorithms (EAs), which have proven effective in tackling optimization problems with high complexity, number of variables and dimensionality [Wang et al., 2010]. DE search for the optimum entails three phases, called mutation, crossover and selection [Storn et al., 1997; Holland, 1975].

In the mutation phase, for each *r*-th bit x_r of the *NP* chromosomes $\bar{x}=(x_1,x_2,...,x_R)$ present in the population at the *g*-th generation, *g*=1,2,…,*G*, an estimation probability $P(x_r)$ is calculated:

$$P(x_r)=\frac{1}{1+e^{-\frac{2b\left[x_r^l+F\left(x_r^k-x_r^m\right)-0.5\right]}{1+2F}}} \tag{1}$$

where *b* is a positive real constant typically chosen ∈[6,9], *F* is a constant user-defined weighting factor typically chosen ∈[0,2] and x_r^l, x_r^k and x_r^m are the *r*-th bit of three randomly chosen chromosomes with indexes $l,k,m\in\{1,2,...,NP\}$. From the probability estimation vector

$P(\bar{x}) = (P(x_1), P(x_2), ..., P(x_R))$, the corresponding bits of the noisy vector $\bar{v}$ of the current chromosome $\bar{x}$ are generated:

$$v_r = \begin{cases} 1 & \text{if } rand \le P(x_r) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where *rand* is a uniform random number in [0,1).

The *r*-th bit of the trial chromosome $\bar{u}$ can be obtained by the crossover operator through Eq. (3):

$$u_r = \begin{cases} v_r & \text{if } rand \le CR \text{ or } r = irand(R) \\ x_r & \text{otherwise} \end{cases} \quad (3)$$

where $CR \in [0,1]$ is a control parameter which influences the probability for each *r*-th bit of $\bar{v}$ to be selected, *irand*(*R*) is a uniform discrete random number from the set $\{1,2,...,R\}$, where *R* is the length of the chromosome (i.e., the number of bits). Therefore, at least one bit of the trial chromosome $\bar{u}$ is inherited from the mutant chromosome $\bar{v}$ so that DE is able to avoid duplication of chromosomes $\bar{x}$ and effectively search within the neighborhood; this contributes to maintaining the diversity inside the perturbed population, shuffling old and new information, and also increases the probability of maintaining some good properties from $\bar{x}$, avoiding drastic changes during the generation of new solution.

During the selection process, the population is modified by substitution. Referring to a minimization search, if the fitness of the trial chromosome $\bar{u}$, i.e., the total cost of all Π belonging to $\bar{u}$, is less than the fitness of $\bar{x}$, the former will be a member of the g+1-th generation replacing the latter, or the latter will be maintained, otherwise:

$$\bar{x} = \begin{cases} \bar{u} & \text{if } fitness(\bar{u}) < fitness(\bar{x}) \\ \bar{x} & \text{otherwise} \end{cases} \quad (4)$$

The fitness used in the DE applications that follow is called "One complement" fitness function [Shackleford et al., 2001]: it accounts for the literal cost of the Π selected to be in the cover and the cost associated with the number of faulty minterms left uncovered. In particular, the cost of the trial chromosome $\bar{u}$ is mapped into a binary number made up by two parts: the most important digits are determined as the complement to one of the uncovered faulty minterms, whereas the least important digits are determined as the complement to one of the sum of the costs of the cut sets included in the trial cromosome. In this way, a complete subset of cut sets that covers all faulty minterms has surely a larger fitness than any other incomplete subset. Moreover, since the selection criterion of DE is greedy, for sure the following generation is better than or at least equal to the previous generation.

3.2 Hierarchical Differential Evolution

The novelty of the Hierarchical Differential Evolution (HDE) here proposed for mcs identification, builds on the setting of a two-steps DE optimization (The pseudocode is shown in Fig. 1). The first optimization is fed with subsets Γ_s, *s=1, 2, ...,S,* of the whole set Ω of cut sets Π, where the *s*-th subset Γ_s is generated by randomly assigning to it *N* cut sets Π of Ω in a way that each cut set belongs to only one subset, i.e., $\Gamma_s \bigcap \Gamma_p = \varnothing$ for $s \neq p$, and the union of all the subsets is equal to all the cut sets, i.e. $\bigcup_{s=1}^{S} \Gamma_s = \Omega$.

For each of the subsets Γ_s, *s=1, 2, ...,S,*

1a) we build a cut set chart, using all the minterms as columns and the cut sets Π belonging to Γ_s as rows

2a) we build the cost vector, where to each Π is assigned its literal cost

3a) we perform the DE optimization (Section 3.1)

4a) we find the best chromosomes $\{\Pi\}_s$.

The second DE optimization is performed on the new subset $\bigcup_{s=1}^{S}\{\Pi\}_s$ comprising all the cut sets included in the best chromosomes $\{\Pi\}_s$ found at the end of the first optimization. In detail,

1b) we build a new cut set chart, using all the minterms as columns and the cut sets belonging to $\bigcup_{s=1}^{S}\{\Pi\}_s$ as rows

2b) we build a new cost vector where to each cut set belonging to $\bigcup_{s=1}^{S}\{\Pi\}_s$ is assigned its literal cost

3b) we perform the DE optimization (Section 3.1)

4b) we find the mcs Π^* of the system.

Three performance indicators are used to judge the goodness of the results. In the evaluation, the optimizations are repeated a number of times (5 in our case), to account for the inherent stochasticity of the search algorithm. The three performance indicators are:

- Cpu: cpu time (expressed in seconds) necessary to converge to the solution $\hat{\bar{x}}_{opt}$.
- Success rate (Sr): percentage of trials for which the true optimum $\bar{x}_{opt}$ is found.
- Accuracy (λ): the larger λ, the larger the accuracy of the solution [Tvrdìk, 2006] as:

$$\text{if } \bar{x}_{opt} \neq 0 \qquad \lambda = \begin{cases} 0 & \text{if } \dfrac{\left|\hat{\bar{x}}_{opt} - \bar{x}_{opt}\right|}{\left|\bar{x}_{opt}\right|} \geq 1 \\ 11 & \text{if } \dfrac{\left|\hat{\bar{x}}_{opt} - \bar{x}_{opt}\right|}{\left|\bar{x}_{opt}\right|} < 10^{-11} \\ -\log_{10}\left(\dfrac{\left|\hat{\bar{x}}_{opt} - \bar{x}_{opt}\right|}{\left|\bar{x}_{opt}\right|}\right) & \text{otherwise} \end{cases}$$

$$\text{if } \bar{x}_{opt} = 0 \qquad \lambda = \begin{cases} 0 & \text{if } \left|\hat{\bar{x}}_{opt}\right| \geq 1 \\ 11 & \text{if } \left|\hat{\bar{x}}_{opt}\right| < 10^{-11} \\ -\log_{10}\left(\left|\hat{\bar{x}}_{opt}\right|\right) & \text{otherwise} \end{cases} \qquad (5)$$

```
for s=1:S
        sample without replacement N cut sets Π from Ω
        populate the s-th subset Γ_s
end
for s=1:S
    create an initial population of NP potential solutions x̄ containing the R cut sets Π belonging to the s-th
         subset Γ_s
    for g=1:G
       select (for each potential solution x̄) three randomly chosen chromosomes for reproduction (Eq. 1)
       create (for each x̄) a noisy vector v̄ using mutation process (Eq. 2)
       create a trial vector ū mixing x̄ and v̄ (Eq. 3)
       compare x̄ with each related trial ū and eventually replace (Eq. 4)
    end
    memorize all the cut sets Π contained in the s-th best solutions set {Π}_s
end
create an initial population of NP potential solutions x̄ composed by {Π}_s , s=1,2,...,S
    for g=1:G
       select (for each potential solution x̄) three randomly chosen chromosomes for reproduction (Eq. 1)
       create (for each x̄) a noisy vector v̄ using mutation process (Eq. 2)
       create a trial vector ū mixing x̄ and v̄ (Eq. 3)
       compare x̄ with each related trial ū and eventually replace (Eq. 4)
    end
memorize the best solution found x̂_opt that contains the mcs Π*
```

First step DE

Second step DE

Fig. 1. *Pseudocode of the HDE optimization technique*

4. APPLICATION TO NUCLEAR SAFETY SYSTEMS

4.1. PWR Reactor Protection System analysis

The procedure for mcs identification developed in Section 3 is here applied to the Reactor Protection System (RPS) of a Nuclear Power Plant (NPP) for the case of a small Loss of Coolant

Accident (LOCA) [Marseguerra et al., 2004]. The RPS is a multi-channel electrical alarm and actuating system that monitors the operation of the reactor. During normal control of the reactor, the rods are raised or lowered into the core by the use of magnetic jacks. Upon detection of an abnormal condition, RPS initiates counteracting actions to prevent a potentially unsafe condition: control rods are rapidly dropped into the core by removing the voltage to the magnetic jacks in order to allow the shutdown of the reactor. More precisely, control rod assemblies are dropped by removal of power through the opening of either the reactor trip breaker of Train A (RTA) or of the reactor trip breaker of Train B (RTB) [Marseguerra et al., 2004]. The two trip breakers, connected in series, control the power provided by two motor generators connected in parallel. Each of them is bypassed by a special test breaker of the same type of the trip breakers, called bypass A (BYA) and bypass B (BYB), for RTA and RTB, respectively. The tripping signals which trip the breakers come from two relay logic trains which are identical in design, called Trip Train A (TrainA) and Trip Train B (TrainB). For prevention against possible interactions that may cause false scrams or failure to scram, the system trips on loss of electrical power and each trip channel is physically separated from the others and from other equipment [Schreiber et al., 2009].

In this analysis, the top event of the RPS FT considered consists in at least 2 out of 48 rods failing to enter the core following a small LOCA, which leads to reactor scram [Wash-1400, 1976]. The failed insertion of the control rod can be originated by a core distortion (CD) (e.g., a change of the channel geometry due to the swelling of the fuel cladding) or by a failure in the rod drop (RDF). Wire faults (WF) are lumped into a single fault and a common mode failure (CMF) is considered, involving several trip circuit breaker faults and wire faults on each branch of the redundant trip breaker system. Trip trains (TrainA and TrainB) can even independently fail to deliver the signal to the RPS trip breakers, while the trip breakers (RTA and RTB) can fail upon receipt of a valid signal (e.g. due to sticking of the undervoltage trip attachment), as their bypass trip breakers (BYA and BYB). Operators interference in the correct automatic operation is considered during testing or maintenance operations (TestA and TestB). Therefore, CD, RDF, WF, CMF, TrainA, TrainB, RTA, RTB, BYA, BYB, TestA and TestB are considered as basic failure events (Tab. 1). The structure function Φ for the considered RPS failure event gives rise to 4096 minterms (4052 leading the RPS into failure state) and 485361 cut sets Π. Further details on the modeling assumptions, the detailed data for the reliability analysis and the FT for the considered top event can be found in [Wash-1400, 1976]. For the sake of clarity, in Fig. 2, a sketch of the complete FR of the small LOCA for the RPS is provided.

	Basic Failure Events	**ID Code**
1.	Trip Train controlling RTA	TrainA

2.	Trip Train controlling RTB	TrainB
3.	Reactor trip breaker controlled by RPS Train A	RTA
4.	Reactor trip breaker controlled by RPS Train B	RTB
5.	Special test breaker bypassing RTA	BYA
6.	Special test breaker bypassing RTB	BYB
7.	RTA undergoing testing	TestA
8.	RTB undergoing testing	TestB
9.	Core distortion	CD
10.	Failure in the rod drop	RDF
11.	Wire failure	WF
12.	Common mode failure	CMF

Table 1. *Basic failure events and failure codes for the RPS system*

The true solution $\bar{x}_{opt}$ that will be used for comparison in the following subsections, has been obtained by traditional consolidated algorithms in [Wash-1400, 1976]: it consists in 15 mcs Π^* ({CD}, {RDF}, {WF}, {CMF}, {RTA, RTB}, {RTA, BYB}, {RTA, TrainB}, {RTB, BYA}, {RTB, TrainA}, {BYA, BYB}, {BYA, TrainB}, {BYB, TrainA}, {TrainA, TrainB}, {TrainA, TestB}, {TrainB, TestA}) [Marseguerra et al., 2004]. Moreover, the results provided by the proposed HDE will be also compared with a DE approach [Di Maio et al., 2013], for showing the improved capabilities of the HDE with respect to similar algorithms.

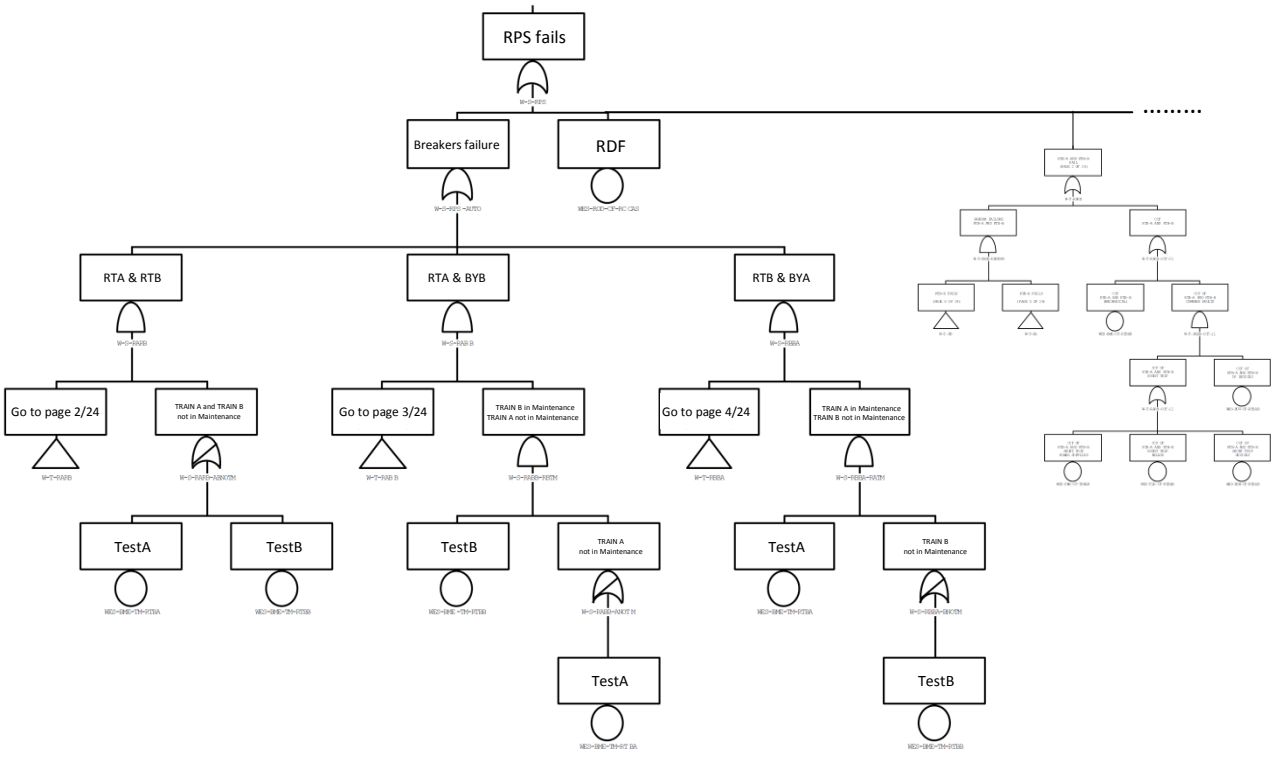

Fig. 2. *Partial reproduction of the complete FT of the of the RPS system* [Wash-1400, 1976]

4.1.1. DE Results

We apply to the RPS case study the DE approach [Di Maio et al., 2013] with a "One complement" fitness function [Shackleford et al., 2001] embedded into the evolutionary algorithm: since the columns in the cut set chart are 4052 (that is, equal to the number of minterms leading the system into failure state) 12 bits code the maximum number of uncovered columns, whereas 22 bits code the literal cost part of the trial solution because the sum of the literal cost of all the 485361 cut sets is equal to 3936648. In Fig. 3, the calculation procedure of the "One complement" fitness function is shown for the best solution $\bar{x}_{opt}$ of the RPS mcs identification problem: in this particular case, the uncovered columns are equal to zero, whereas the total cost of the best solution $\bar{x}_{opt}$ is equal to 4·1+11·2=26 (i.e., 4 cut sets contain only one basic event and 11 contain 2 basic events); the complement to one of 0 on 12 bits is equal to 4095, and the complement to one of 26 on 22 bits is equal to 4194277; joining together the two parts of the fitness function gives a fitness value for $\bar{x}_{opt}$ equal to 17179869157.

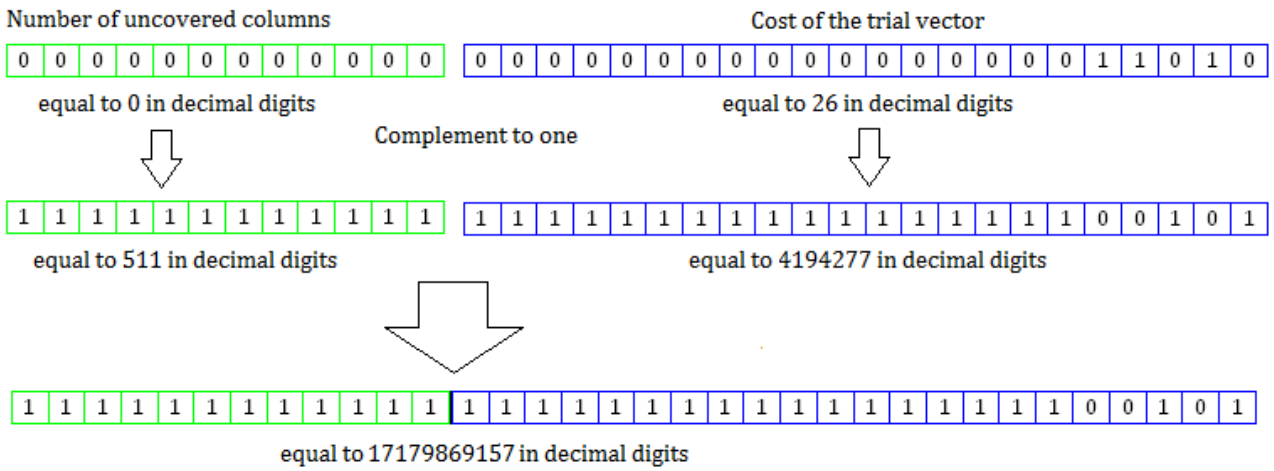

Fig. 3. *Procedure for the calculation of the fitness function for the best solution of the RPS system*

In this application, parameters *F* and *b* (Eq. 1) and *CR* (Eq. 3) are set equal to the values reported in Tab. 2.

Parameters	***F***	0.1
	b	9
	CR	0.2

Table 2. *Values of the parameters F, CR and b used in the DE*

The analysis is performed for a population size *NP*=50, because this is the maximum value allowed by the computer memory constraints: the identification of the mcs for the RPS entails chromosomes

of length R=485361 (i.e., the number of cut sets Π), and an allocated memory of $NP \times R$ bits (equal to 24268050, in our case); any further increase in NP is not allowed by the Matlab® software used in this work to develop the DE and HDE. The only stopping criterion is the generation number G set equal to $MAXGEN$=10000. Performance indicators are the same introduced in Section 2 and are quantified on a set of 5 trials of optimization. Results of the DE optimizations are shown in Tab. 3.

NP	50
Cpu [s]	91157.80
Sr	0 %
λ	2.16

Tab. 3. *Performance indicators for the DE-based algorithm, with NP=50*

It is seen that in this real case with a large number of minterms and cut sets, DE is not capable of finding the true solution $\bar{x}_{opt}$ (found in [Wash-1400, 1976] by traditional consolidated algorithms) among all the cut sets (Sr=0%), due to the hardware computational capability that limit the population size to NP=50. To overcome this limitation, we apply HDE to the FT of the RPS.

4.1.2. HDE Results

In accordance with the procedural steps presented in Section 3.2 and Fig. 1, we partition Ω into S=50 subsets Γ_s (39 subsets composed by 9707 cut sets and 11 subsets composed by 9708 cut sets): the number of the cut sets belonging to each Γ_s is chosen guided by the fact that the DE has shown good results (in terms of success rate (Sr)) when applied to cut sets groups smaller than 10000 [Di Maio et al., 2013]. For each Γ_s, its cut set chart and its cost vector are built as shown in Section 3.2. When parameters F, b, CR are set as in Table 2, NP=100 and $MAXGEN$=1500, the total time approximately required for the first level optimization is 25721 [s] on an Intel® Core™ i5.2500 CPU @3.30GHz.

The number of cut sets Π found by the first step of the optimization is 1510, among which the second step DE will search for the mcs Π^*. For this, the cut set chart and the cost vector associated to the new cut sets are defined as shown in Section 3.2. As for the first optimization step, the parameters F, b, CR are set equal to the values reported in Tab. 2, NP=100 and $MAXGEN$=1500. The mcs found by the HDE are the same as those reported in [Marseguerra et al., 2004], proving the HDE procedure effective in finding the mcs in large structure functions. The cpu time required for the second-step optimization is equal to 990.71 [s]. Thus, the total time required by the HDE optimization is equal to 26700 [s], much shorter than for the DE (Table 3).

For showing the better results of HDE with respect to DE, in Fig. 4, the evolution of the difference Δ between the fitness values of $\bar{x}_{opt}$ and $\hat{\bar{x}}_{opt}$ is shown on a semi-logarithmic plot. HDE shows superior convergence performance. In fact, in the first step of the optimization (continuous line with circles) it achieves better results than DE (continuous line with triangles) by resorting to a larger population for exploring a reduced and focused search space, whereas in the second step of the optimization (continuous line with stars) it explores an even more reduced search space made up of the selected best chromosomes $\bigcup_{s=1}^{S}\{\Pi\}_s$ reaching Δ=0 in only 2200 generations. It is worth pointing out that, with respect to the accuracy of the solution found and the success rate of HDE (for a set of 5 trials), at the end of the first optimization stage it is meaningless to calculate Sr and λ, because the true solution $\bar{x}_{opt}$ might not be included into the set of solutions $\{\Pi\}_s$ of the S subsets Γ_s. On the other hand, at the end of the second optimization step, HDE provides Sr=100% and λ=11.

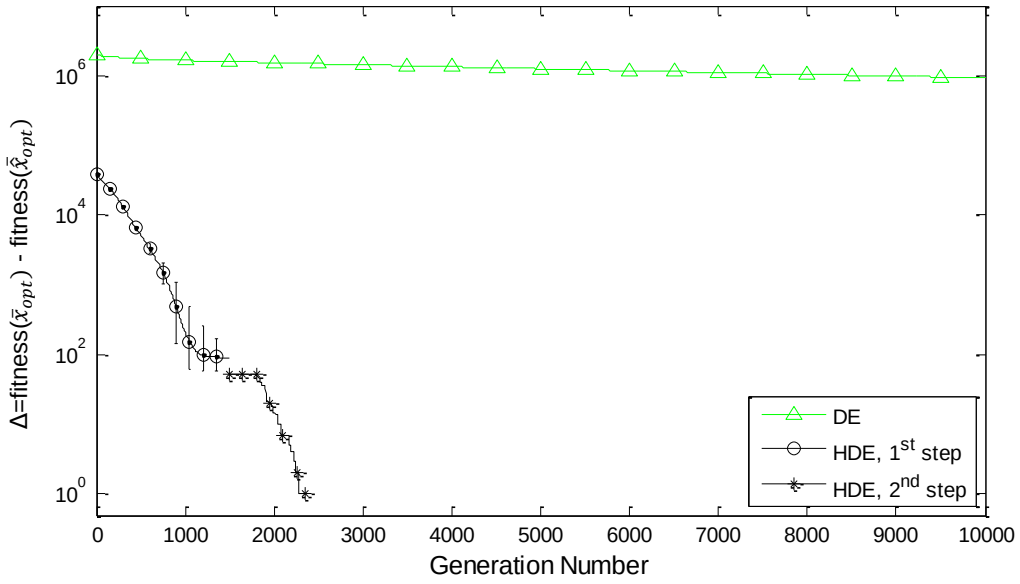

Fig. 4. *Fitness function convergence using ordinary DE and HDE*

In principle, the extension to a hierarchical, multi-step, DE-based algorithm for mcs identification is straightforward, allowing the treatment of very large systems. It is always feasible to group the possible solutions in different subgroups and then run a different optimization for each subgroup until the solution converges to the optimum of the fitness function. In other words, applying a hierarchical DE-based procedure, the NP-complete problem [Sen, 1993] associated to the mcs identification of a complex (even non-coherent) structure function can be tackled in such way that

the computational complexity of the problem grows linearly with the number of subgroups dimension and not exponentially, as it is when resorting to the single DE optimization.

4.2. CANDU Airlock System

A second application of the HDE for mcs identification considers the FT developed for analyzing a scenario of a Design Basis Accident (DBA) occurred in 2011 in the Airlock System (AS) of a CANDU NPP [Lee et al., 2012; Di Maio et al., 2013b]. The AS is a safety system required to keep the pressure of the inner side of the reactor vault lower than the outer side in order to avoid the dispersion of contaminants out of the reactor bay, in case of accident. Therefore, the FT top event is the incapability of the AS to maintain the pressure boundary [Lee et al., 2012]. The system consists of a vessel in the containment wall of the reactor vault, with two doors in order to allow the inspection of the vault: one door opens towards the inside of the reactor vault, the other towards the outside; so, at least one airlock door, whose seals are normally inflated via the air system, must be closed by a latch with sufficient pressure in the seals to fulfill its safety function. During the accident, the inflation of the seals is switched to the back-up air supply tank. Possible causes for the top event occurrence can be: the pressure equalizer valve fails (V1), doors fail to close because latches are not locked (D1) and seals are cracked or cannot be inflated (S1). The pressure equalizer valves are designed to equalize the pressure between the reactor bay and the service side and, therefore, to allow controlled flow between these two areas. The pressure equalization can fail due to gear box failure (G1) that may limit the vents from opening and closing, to the presence of leakages in the piping system (P1/P2) or to the failure of the exhaust pipe (E1). The airlock doors must be closed by a latch, otherwise the pressure equalizer valves and seals cannot be called in operation on demand. In addition, the possibility is considered that the back-up tank is already empty (T1) or fails to engage (T2) when the inflation of the seals is switched to the back-up air supply system. The basic failure events that can give rise to the AS failure are listed in Table 6.

	Basic Failure Events	**ID Code**
1.	Pressure equalizer valve is failed	V1
2.	Doors fail to close and lock	D1
3.	Seals are cracked	S1
4.	Gearbox fails	G1
5.	The piping system presents minor leakages	P1
6.	The piping system presents major leakages	P2
7.	Exhaust pipe fails open	E1
8.	Back up tank is empty	T1
9.	Back up tank fails to engage	T2

Tab. 6. *Basic failure events and failure codes for a DBA in a CANDU AS* [Di Maio et al., 2013b]

The FT for the DBA here considered is shown in Fig. 5 [Lee et al., 2012]. The structure function expression is $\Phi = \left[\left(G1 \text{ and } E1 \right) \text{ or } \left(\left(T1 \text{ and } \left(S1 \text{ or } V1 \text{ or } P1 \right) \right) \text{ or } \left(V1 \text{ or } T2 \text{ or } P2 \right) \right) \text{ or } D1 \right]$. There are 497 minterms leading to the system failure, 16867 cut sets and 7 mcs Π^*=({D1}, {P2}, {T2}, {V1}, {E1,G1}, {P1,T1}, {S1,T1}) as found in [Lee et al., 2012] by traditional consolidated algorithms for mcs identification.

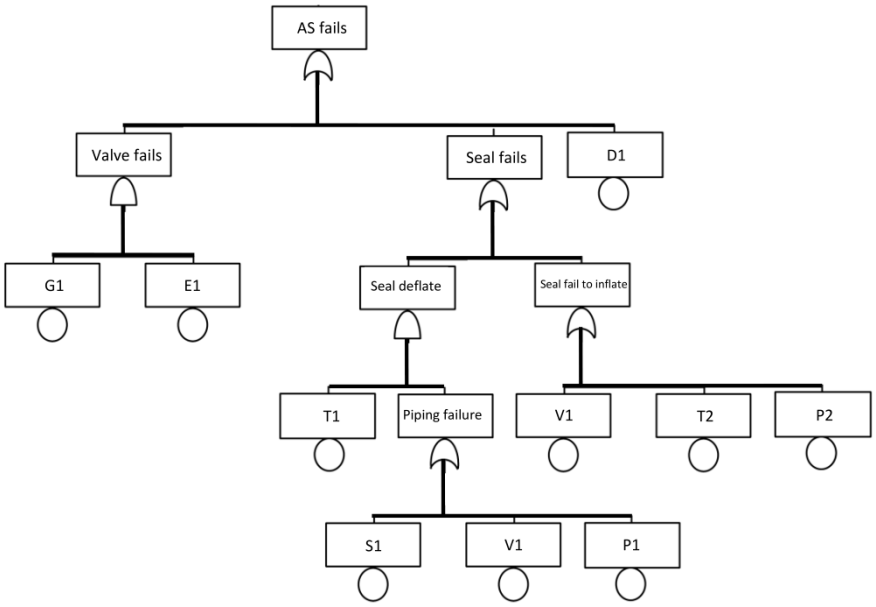

Fig. 5. *FT for the DBA of the AS* [Lee et al., 2012]

4.2.1. HDE Results

In Fig. 6 the calculation procedure of the "One Complement" fitness function is shown for the best solution: in the problem of the AS of the CANDU, where the columns of its cut set chart are 497 (that is, equal to the number of minterms), 9 bits code the maximum number of uncovered columns, whereas the sum of the cost of all the 16867 cut sets is equal to 103298 so that 17 bits code the cost part of the trial solution. The uncovered columns are equal to zero, while the total cost of the best solution is equal to 10 (4 cut sets contain only one basic event and 3 contain 2 basic events); the complement to one of 0 on 9 bits is equal to 511, and the complement to one of 10 on 17 bits is equal to 131061; joining together this two parts of the fitness function gives a fitness value for $\bar{x}_{opt}$ equal to 67108853.

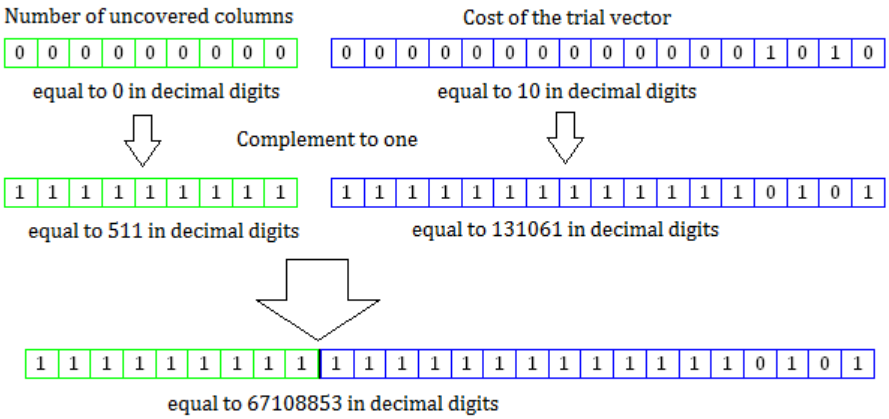

Fig. 6. *Procedure for the calculation of the fitness function for the best solution of the CANDU AS*

In the proposed two-step HDE framework, we have set the number of subsets Γ_s equal to S=10 and the number of minimal cut sets Π of Ω equal to N=1687 for each s-th subset Γ. We perform the first-stage DE optimization with a population size equal to NP=500 and stopping criterion "maximum number of generation", $MAXGEN$=700. The mean cpu time required for performing the optimization on a single s-th subset is equal to 607.20 [s]. At this first optimization stage, the number of cut sets is 100 and the true solution $\bar{x}_{opt}$ does not belong to any of the S subsets. However, it is worth pointing out that on Intel® Core™ i5.2500 CPU @3.30GHz the computational demand approximately required for the first step is 1214.40 s.

We perform the second DE optimization comprising all the 100 cut sets included in the best individuals $\{\Pi\}_s$ found at the end of the first optimization. The population size equal to NP=500 and $MAXGEN$=200. The mcs found by the HDE are the same as those reported in [Lee et al., 2012], proving that the HDE is capable of identifying the exact mcs Π^* of a complex system as the AS of a CANDU. The cpu time required for the second step optimization is equal to 19.76s making the total time required by the HDE optimization equal to 1233s.

For further comparison of the results with other similar algorithms, in Fig. 7 the faster convergence obtained by the HDE compared with DE with NP=700 is shown: the faster evolution towards zero of the difference Δ between the fitness values of $\bar{x}_{opt}$ and $\hat{\bar{x}}_{opt}$ of HDE highlights its superior performance. In fact, in the first step of the optimization (continuous line with circles) it achieves better results than DE (continuous line with triangles) by resorting to a larger population for exploring a reduced search space, whereas in the second step of the optimization (continuous line with stars) it explores an even more reduced search space made up of best individuals, reaching Δ=0 in only 800 generations, whereas around 1550 generations are needed for DE such that Δ=0.

As a final remark, it is worth pointing out that, in this latter case, *i*) the number of generations needed for Δ=0 is much smaller than for the RPS case study of Section 4.1.2 and *ii*) DE is still capable of finding the solution $\bar{x}_{opt}$ of the mcs identification problem, whereas in Fig. 7 the unfeasibility of resorting to DE for the RPS case study is clear. This is due to the fact that the number of components of the RPS case study is larger than that of the AS here considered.

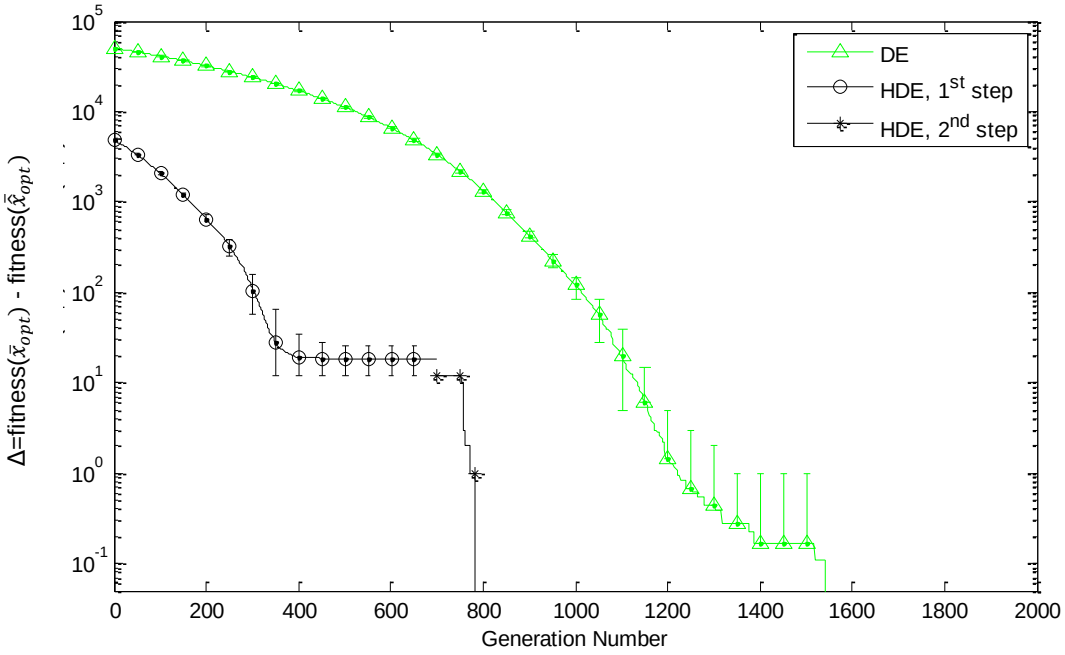

Fig. 7. *Fitness function convergence using DE and HDE*

5. CONCLUSIONS

The exact identification of the mcs of FTs is an important task in PSA. It becomes non-trivial for systems that are composed by large numbers of components. In this paper, we have addressed this issue by proposing a novel HDE algorithm. This amounts to transferring the mcs identification into a hierarchical optimization problem: during the first step, a multiple-population, parallel DE search policy is used to expedite the convergence of a second step of DE exploration. The proposed method has been applied for the analysis of a RPS of a PWR and a AS of a CANDU. The superior HDE performance is evident when the number of basic events in the FT is large.

References

Akers, B. (1978). Binary decision diagrams. IEEE Transactions on Computers, 276, 509-516.

Beasley, J.E., Chu, P.C. (1996). A genetic algorithm for the set covering problem. European Journal of Operational Research, vol.94, 392-404.

Bjorkman, K. (2013). Solving dynamic flowgraph methodology models using binary decision diagrams. Reliability Engineering and System Safety, 111, 206-216.

Borgonovo, E. (2010). The reliability importance of components and prime implicants in coherent and non-coherent systems including total-order interactions. European Journal of Operational Research, 204, 3, 485-495.

Di Maio, F., Baronchelli, S., Zio, E., (2013). Prime Implicants Determination by Differential Evolution for Dynamic Reliability Analysis of Non-Coherent Systems. under review, Reliability Engineering and System Safety.

Di Maio, F., Baronchelli, S., Zio, E. (2013b). Minimal Cut Sets Identification by Hierarchical Differential Evolution. PSA 2013, the International Topical Meeting on Probabilistic Safety Assessment and Analysis, 22-27 September 2013, Columbia, South Carolina, USA.

Duflot, N., Bérenguer, C., Dieulle, L., Vasseur, D. (2009). A min cut-set-wise truncation procedure for importance measures in probabilistic safety assessment. Reliability Engineering and System Safety, 94, 1827-1837.

Epstein, S., Rauzy, A. (2005). Can we trust PRA?. Reliability Engineering and System Safety, 88, 195-205.

Fleming, K.N. (2003). Issues and recommendations for advancement of PRA technology in risk-informed decision making. Technical Report NUREG/CR-6813, U.S. Regulatory Commission.

Gao, X., Cui, L., Li, J. (2007). Analysis for joint importance of components in a coherent system. European Journal of Operational Research, 182 (1), pp. 282-299.

Garrett, C., Guarro, S., Apostolakis, G. (1995). The dynamic flowgraph methodology for assessing the dependability of embedded software systems. IEEE Transactions on Systems, Man and Cybernetics, 25, 824-840.

Holland, J.H. (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor.

Høyland, A., Rausand, M. (1994). System Reliability Theory: Models and Statistical Methods. John Wiley & Sons.

Kumamoto, H., Henley, E.J. (1996). Probabilistic risk assessment and management for engineers and scientists. New York, IEEE Press.

Labeau, P.E., Smidts, C., Swaminathan, S. (2000). Dynamic reliability: towards an integrated platform for probabilistic risk assessment. Reliability Engineering and System Safety, 68, 219-254.

Lee, A., Lu, L. (2012). Petri Net Modeling for Probabilistic Safety Assessment and its Application in the Air Lock System of a CANDU Nuclear Power Plant. Procedia Engineering, 2012 International Symposium on Safety Science and Technology, Volume 25, pp.11-20.

Marseguerra, M., Zio, E. (2004). Monte Carlo estimation of the differential importance measure: application to the protection system of a nuclear reactor. Reliability Engineering and System Safety, V. 86, 11-24.

NASA (2002). Fault Tree Handbook with Aerospace Applications.

NUREG (1983). PRA Procedures Guide, Vols 1&2, NUREG/CR-2300.

Petersen, J.L. (1981). Petri net theory and the modeling of systems. Englewood Cliffs, NJ: Prentice-Hall.

Rauzy, A., Dutuit, Y. (1997). Exact and truncated computations of prime implicants of coherent and non-coherent fault tree. Reliability Engineering and System Safety, 58, 127-144.

Rauzy, A. (2001). Mathematical Foundations of Minimal Cutsets. IEEE Transactions on Reliability, Vol. 50, No. 4.

Shackleford, B., Snider, G., Carter, R. J., Okushi, E., Yasuda, M., Seo, K., Yasuura, H. (2001). A High-Performance, Pipelined, FPGA-Based Genetic Algorithm Machine. Genetic Programming and Evolvable Machines, Volume 2, Number 1, 33-60.

Schreiber, R., Theriault, K. (2009). Pressurized Water Reactors (PWRs) and Boiling Water Reactors (BWRs). Nuclear Engineering Handbook, K.D. Kok (editor), CRC Press.

Sen, S. (1993). Minimal cost set covering using probabilistic methods. Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing: states of the art and practice, 157-164.

Storn, R.; Price, K. (1996). Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization, 11: 341–359.

Tvrdìk, J. (2006). Competitive differential evolution. MENDEL 2006, 12th International Conference on Soft Computing, 7-12.

Wang, L., Fu, X., Menhas, M.I. (2010). A Modified Binary Differential Evolution Algorithm. Life Modelling and Intelligent Computing, Lecture Notes in Computer Science, Volume 6329.

Wash-1400 (1976). NUREG 75/014, Reactor safety study: an assessment of accident risks in US commercial nuclear power plant, Appendix 2: Fault Trees.

Zio E. (2007). An introduction to the basics of Reliability and Risk Analysis. World Scientific Publishing, 2007.

Zio, E., Di Maio, F., Tong, J. (2010). Safety Margins Confidence Estimation for a Passive Residual Heat Removal System. Reliability Engineering and System Safety, 95, 828-836.